EMBEDDED COMPUTER SOFTWARE LOADER/VERIFIER

IMPLEMENTATION USING A HARDWARE AND SOFTWARE ARCHITECTURE BASED UPON BEST PRACTICES DERIVED FROM MULTIPLE SPIRAL DEVELOPMENTS AND THE JOINT TECHNICAL ARCHITECTURE

24 October 2002

Dan Christenson, MSCS
Chief, Weapon System Software Engineering Branch

Program Information

Ogden Air Logistics Center

Innovation and Excellence

Program Management Office

- WR-ALC/LEACB
- Responsible for:
  - Program Planning and Control
  - Budget and Execution

AAG Development / Program Integration

- OO-ALC/MASMD
- Responsible for:
  - AAG Development
  - System Integration
  - Field Support
  - Production
  - Information Management

Topics

System Introduction

- Overview, Challenges & History
- Constraints & Architecture

Hardware/Software

- Additional Constraints & Criteria
- Requirements Allocation
- Management of Implementation
- Pitfalls

System Wrap-up

- System Relationship
- Success

Overview

A SYSTEM… Using a Portable Rugged Computer to Reprogram Aircraft Avionics

- Provide Universally Common Reprogramming Capability
  - All Air Force Systems
  - Generic Commercial Products & Technology
- Replace Multiple Loader / Verifiers on Multiple Aircraft
  - Retain Functionality of All
- 1600 CAPRE Systems and 55 Aircraft Adapter Groups (AAGs)

IS USED… Backshop & Flightline to Download & Transfer Mission Critical Data (MCD)

- Operational Flight Programs (OFPs), Mission Data Files (MDFs), etc.

SUPPORTS… wide variety of weapon systems

- 32 Rotary and Fixed Wing Aircraft Weapon Systems
- 258 Weapon System to LRU Interfaces

PROVIDES... weapon system updates for capability improvements / corrections

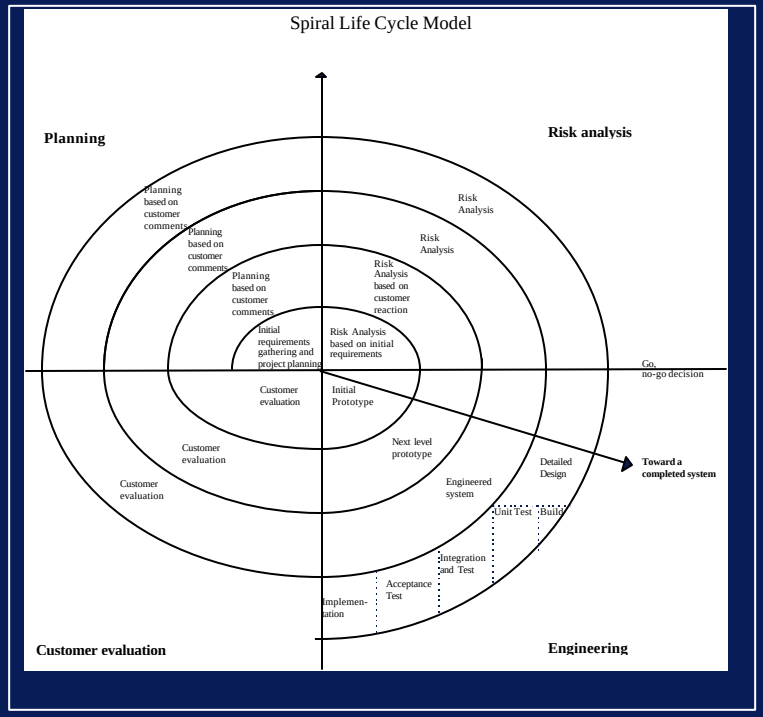
Challenges

Functional Replacement
- New Technology Rehost
- System Characterization

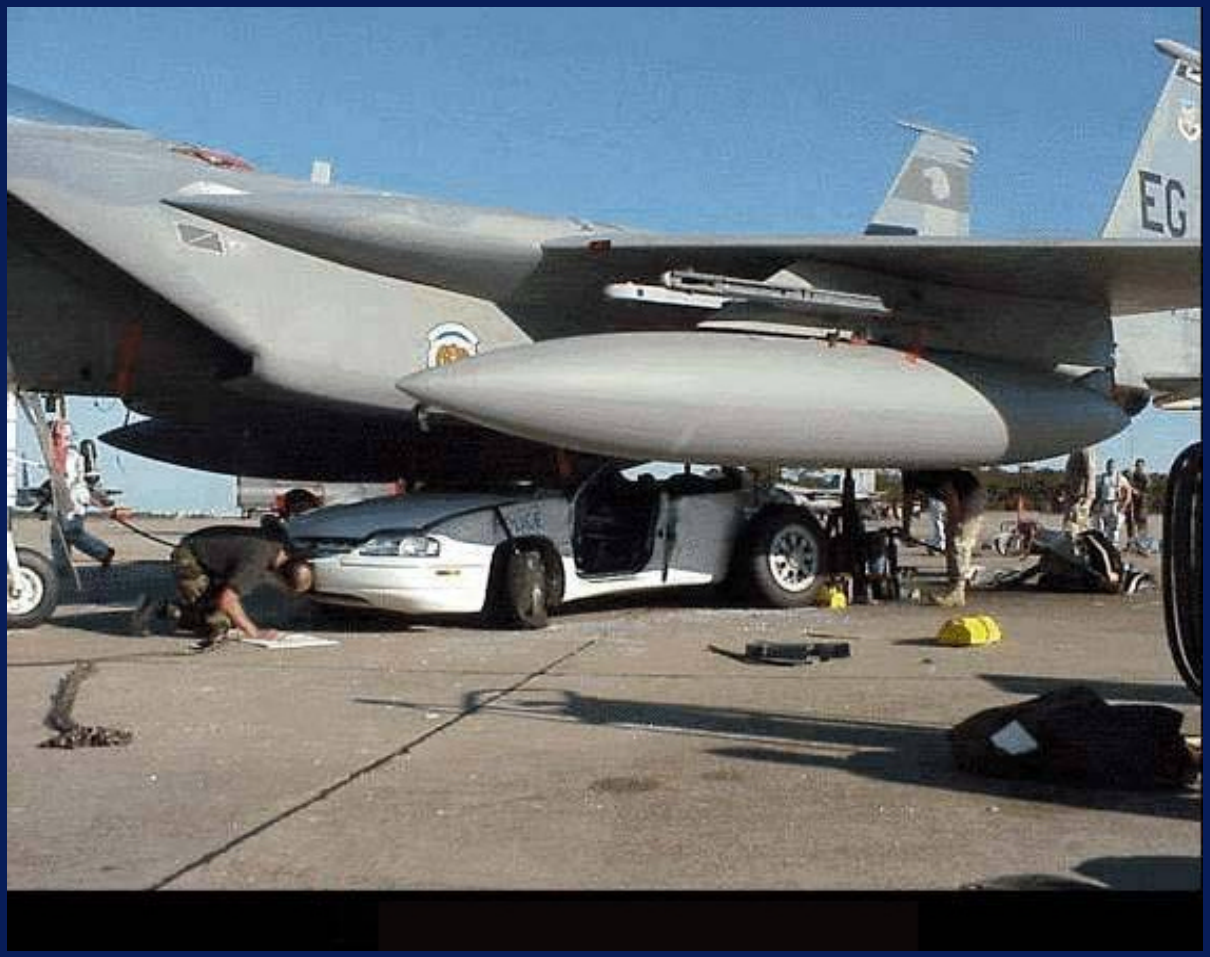
Long Term Supportability
- COTS / Military Interface
  - Forecast Market & DoD Directions
  - Custom Stable Interface
- Dynamic Environment
  - Platform and O/S Independence
  - Technology Refresh
  - Extensibility & Scalability

Spiral Life Cycle
- Requirements Derivation & Customer Approvals
- Rapid-fire Successes

Small Co-Development Teams
- Diversity
- Senior Engineer Mentors



Form, Fit, and Function



History

Digital Computer System (DCS)

- Old Technology
- Parts Unavailable
- DOS Operating System
- Limited Repair Capability

Program Loader Verifier/Memory Loader Verifier (PLV/MLV)

- Old Technology
- Mechanical Tape Transfer Units
- Parts Unavailable
- Big & Heavy
- Limited Repair Capability

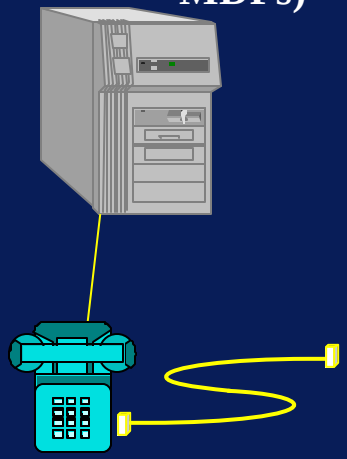
Requirements Implementation Problems

- Custom Computers / Single Platform
- Major Software Changes for O/S or Hardware Changes
- Computer Interface Protocol

Legacy

Step 1

Secure Bulletin Board Server (OFPs and MDFs)



Secure Telephone Unit (STU)

DCS

DCS Downloading OFPs or MDFs

Step 2

DCS **PLV/MLV**

DCS Programming PLV/MLV

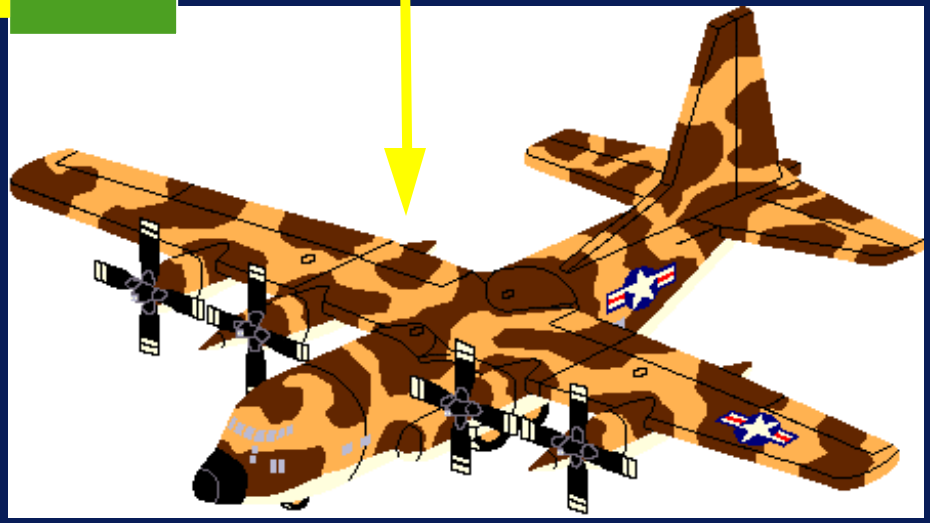
Legacy

Step 3

Aircraft Adapter Group (AAG)

Interface Card (Plugs into PLV/MLV)

PLV/MLV

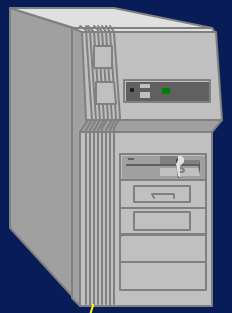


PLV/MLV Reprogramming
Aircraft Avionics

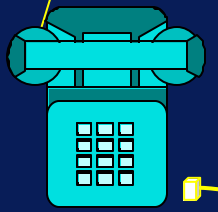
CAPRE

Ogden Air Logistics Center
Innovation and Excellence

Step 1



Secure Bulletin Board Server (OFPs and MDFs)



Secure Telephone Unit (STU)

CAPRE

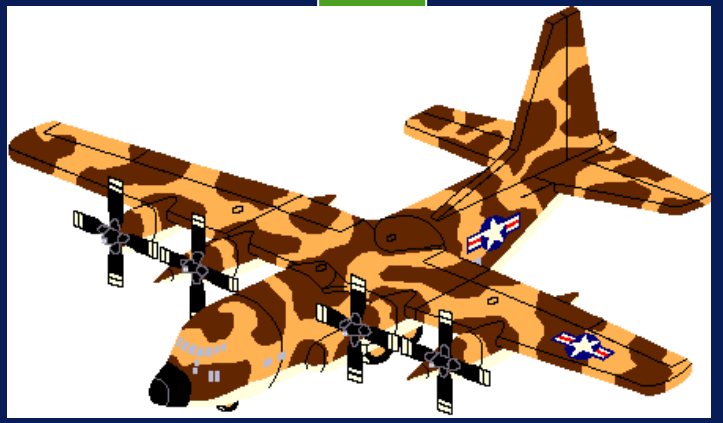
CAPRE Downloading MCD

Step 2

CAPRE

USB Interface

AAG



CAPRE Reprogramming Avionics MCD

System Level Constraints

Deploy as Soon as Possible
- Replace DCS Immediately

Minimize Cost
- Nothing New Here

No Proprietary Data
- Open Source Development

Independent of Specific Manufacturing
- Common Practices

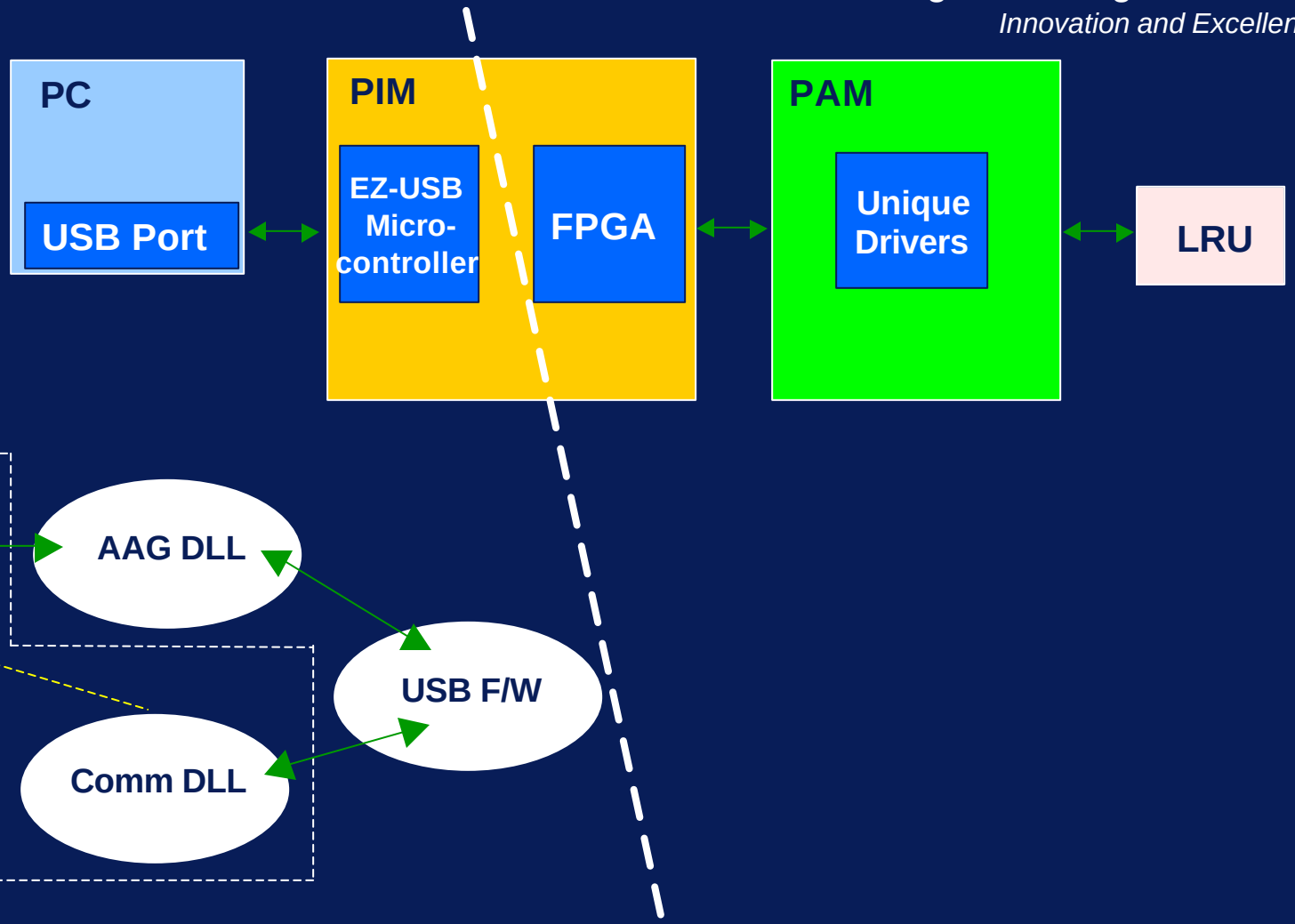
Rugged COTS Portable Computer
- COTS w/ Periodic Replacement

Removable Hard Drive
- Security

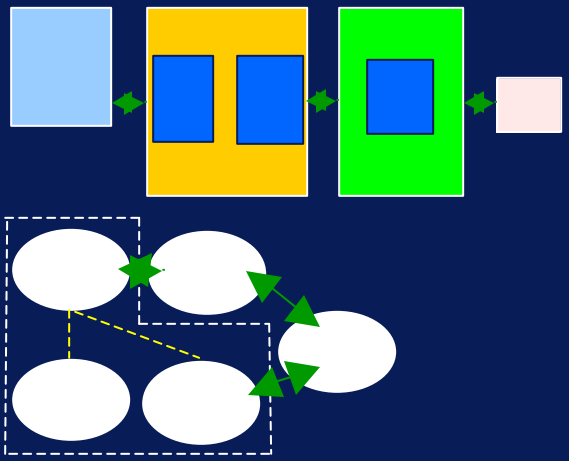
Data Driven Design
- Focus on Each Actor's Added Value

No Changes to MCD Files
- Processing Shall Not Alter Content

Architecture



Hardware and Software Relationships



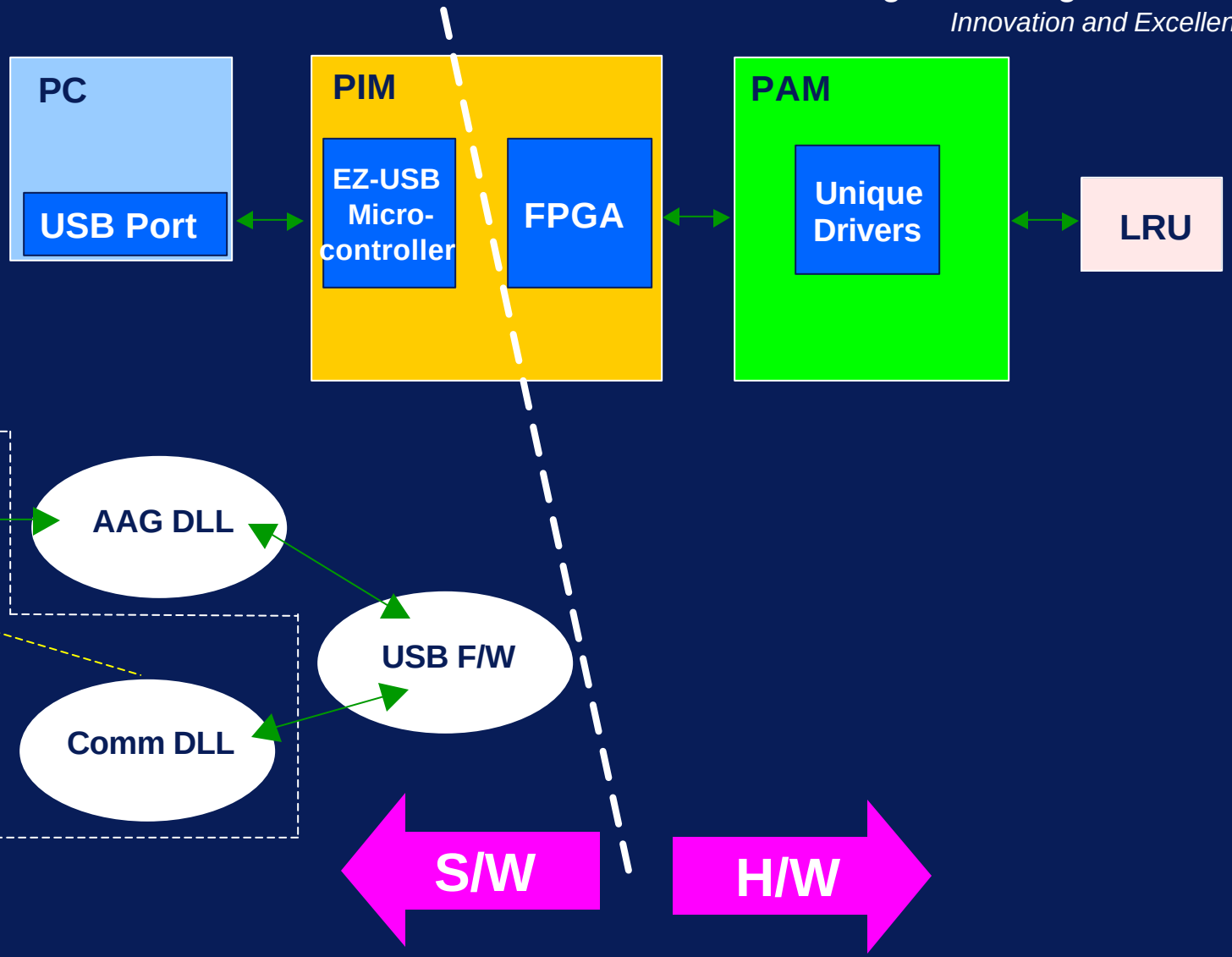
Interfaces

- Rigid Interfaces at Each End
  - MCD Files to AAG DLL
  - LRU to FPGA
- Defined Flexible Interfaces Between
  - User to GUI
  - GUI to Control
  - Control to Database, to Comm DLL, & to AAG DLL
  - AAG DLL to USB Firmware
  - USB Firmware to FPGA

Data Flow Value Added

- Maintain System View
  - Look Both Ways
- Interactions and Functions Identified for Each Actor
  - Additional Requirements Evaluated for Impact to All Actors

Architecture



Scope

Hardware

- All Physical Aspects of AAG
- FPGA Design

Software

- Host Software (Main Executable and GUI Functions)
- AAG DLL (Most AAG Unique Functions)
- USB F/W (High Speed AAG Unique Functions & USB to FPGA Interface)

Software and Hardware must be considered together

- Example: Register order and bit order causes extra writes and bit manipulation.
  - If bits in a data word in the FPGA are reversed from the bit order in the original MCD file the software must move each bit individually into the right order. In the FPGA it is just a matter of defining a register to be MSB vs. LSB.
  - If registers are organized the same way the data is, then multiple registers can be filled with one firmware function. If the registers are out of order then each register must be filled one at a time causing significant software overhead and additional bus traffic.

Additional Hardware Constraints and Criteria

Constraints

- Size
  - FPGA Gates
  - EZ-USB Memory
- Timing
  - USB Bulk Transfers Required for Efficiency
  - 3 ms Minimum Latency on USB Bus
  - Windows Interrupts Can Last More Than 100 ms

Criteria

- Timing
  - AAG DLL Greater Than 100 ms
  - USB F/W 10 to 100 ms
  - FPGA Less Than 10 ms
- Error Handling
- Ease of Implementation
- Stability

Additional Software Constraints and Criteria

Constraints

- No Changes to MCD Files
- No Changes That Would Impact AAGs Already Developed
- USB F/W Code Size and Complexity
- USB Timing (3 ms Turn-around)
- USB Handshaking
- Requirements of All AAGs Not Known

Criteria

- Host Software
  - Common / Shared User Functionality
  - Release Cycle
  - Minimal Changes to Existing
  - Maximal Use of GUI
- CAPRE S/W Library
  - Common / Shared AAG-specific Functionality (e.g. I/O)
  - Complements C++ Standard Library
- AAG S/W & F/W
  - Unique Functionality

Hardware/Software Requirements Allocation

Flexibility is Key to Success

- Requirements Changes Easier to Implement
- Design Effects of Documentation Deficiencies are Easier to Correct
- Design More Likely to Exceed Requirements
- Rapid Prototyping Possible
- Rapid Response to Unexpected Environments
- Process Improvements Implemented Quicker
- Reuse of Existing Interfaces/designs More Likely and Encouraged

Functions Allocated Right to Left

- Fixed LRU interface, FPGA & Cable, Firmware, AAG DLL, and Host Software
- Flexibility Increases As Functions Moved to the Left

MCD Data Flow Allocation Left to Right

- Host Software, AAG DLL, Firmware, and FPGA & Cable
- Data More Easily Manipulated in Software

Example: Automatic buffered 4 channel RS-422 interface wouldn't fit into the FPGA.

- Changed to register driven 4 channel RS-422 and moved handshaking, buffering and error handling into firmware.
- Impact to schedule less than 2 weeks.

Management of Implementation

Management by Walking Around

- Reduces Documentation and Paper Work
- Allows Flexibility
- Errors Found Sooner
- Early Feed Back and Small Changes

Mentors for New Engineers

- New Engineers Learn Process While Contributing
- Less Need for Formal Training
- New Engineer Is the Back-up
- Less Impact Due to Personnel Changes
- Automatic Continuous Mini Peer Reviews

Small Engineering Cells

- Each Cell Responsible for All Aspects of Development
- Each Cell Member was Responsible for Their Cell's Results
- Subject Matter Experts Made Available to Cells
- Cells Isolated from Failures in Other Cells
- More Efficient Peer Reviews
- Cross Pollination Required to Insure Uniform Product

Pitfalls

Late Design Flaw Detection Might Result From Too Much Flexibility

- Subject Matter Expert Vigilance

Late Requirements Changes Might Cause Bigger Schedule and Cost Impacts

- Weekly Status Reports to Customer

Engineers Are Tempted to Go Directly to Implementation and Skip Design Phase

- Develop and Review Documentation as Development Progresses

Process More Complicated and Not Easily Defined

- Subject Matter Expert Mentoring

Mentoring Requirement Slows Growth Potential

- Limit Growth to Four New Engineers per Quarter

More Difficult to Analyze Cause and Effects

- Qualitative vs. Quantitative Analysis

Hardware and Software Relationships

Maintain System View

Success depends on same factors

- Know Your Constraints
  - "A man's got to know his limitations", Clint Eastwood
- Be Flexible in Requirements Allocation
- Define and Insist on Meeting Interfaces
- Focus on Value Added by Each Actor
- Ride Herd on All the Cats

Field Success

Two years into six year program

Deliveries Began in FY01

- 650 in supply/field
- 80% of initial buy fielded
- 17% in supply and available for release to field unit level
- 3% (30 ea) CAPRE units supporting research and development efforts

Developed 10 AAGs Supporting 98 Weapon System / LRU Combinations

Developing 11 AAGs Supporting 28 Weapon System / LRU Combinations

ACC	**51**
PACAF	**25**
USAFE	**39**
AMC	**34**
AFMC	**43**
AETC	**24**
ANG	**144**
AFSOC	**53**
AFR	**38**
Other	**8**

System Success

Platform Independence

- Panasonic CF-27, CF-28, & CF-M34
- Dolch NotePAC
- Paravant Scorpion
- AMREL ROCKY II Plus
- GETAC Series A-320
- EDNA

Operating System Independence

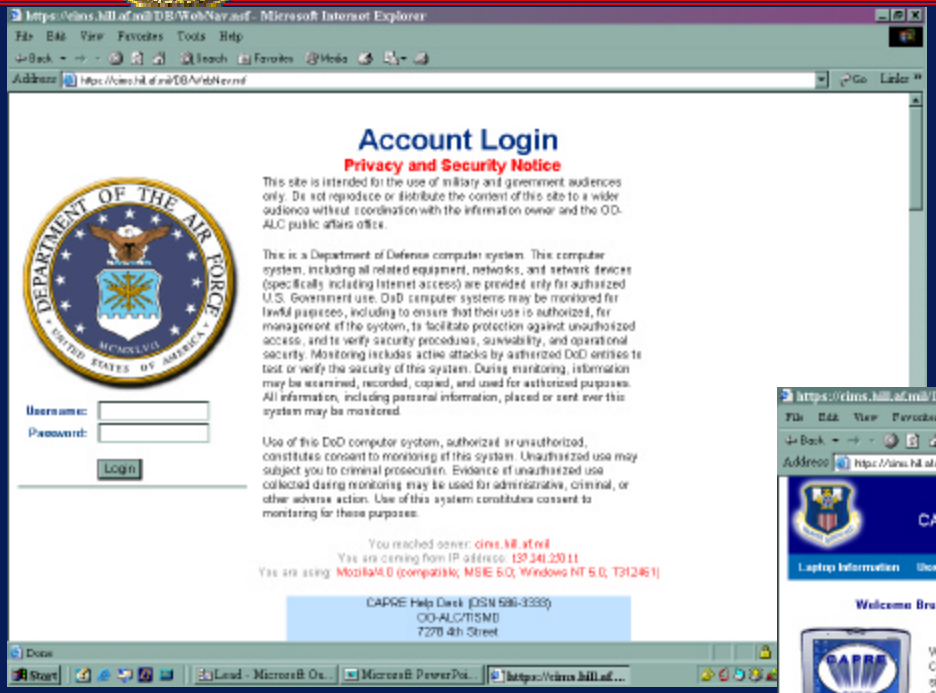
- Windows 98
- Windows 2000
- Windows XP

CAPRE

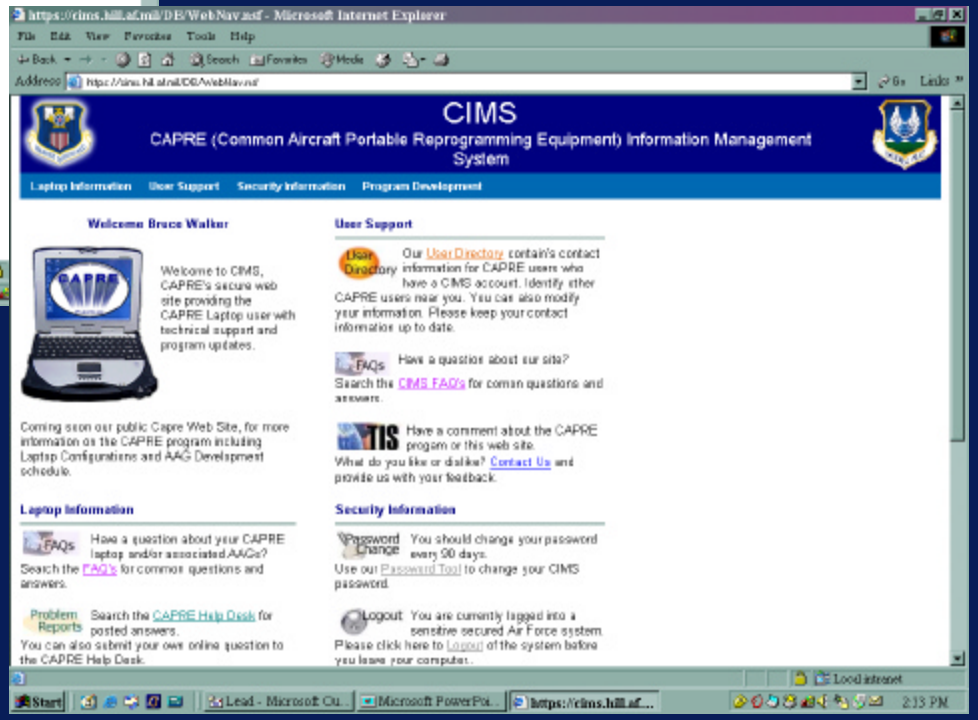
CAPRE SYSTEM

World Wide Web Access

Ogden Air Logistics Center
Innovation and Excellence



https://cims.hill.af.mil



http://capre.hill.af.mil

Back-up Charts

Following Slides
are Available
to Aid Discussion
if Needed

No Unique PC

Past Loader / Verifiers Included Custom Computers

- Custom Cards Installed
- Custom Interfaces Available Only on Specific Computer
- Custom Computers to Meet Too Stringent of Environmental Requirements
- Mechanically Customized Flimsy Standard Interfaces (e.g. PCMCIA)

Solution

- All Custom Electronics External
- High Speed Communication Standard That Is Available Most Computers
- Communication Standard That Provides Power to Custom Electronics
- Tailor the Environmental Requirements
  - Based on End User's Needs and Market Availability

Multiple Platforms

Problem: Loader / Verifier Shall Operate on Multiple Platforms

- Internal Interfaces Handle Timing
- Software Tied to Specific Hardware Due to Bypassing Built-in Drivers
- Non-standard Calls to Display Data
- Third Party Software Requiring a Fee for Each Installation

Solution:

- System Level Solution
- Hardware and Software Co-development
- External Hardware or Standard Operating System Handle Timing Issues
  - No Time Critical Functions Implemented in Software
- Standard Interface With Built-in Operating System Drivers
  - Fast Enough to Handle All Expected Data Transfer Rates
- Standard Operating System Calls or Library of Common Functions
  - Standardizes Multiple Software Developments
- No Third Party Software That Add Fees to Software Distribution

Minimize Software Changes

Problem: Major Software Changes Due to Operating System or Hardware Changes

- Non-standard calls used that may not be available in the next operating system or may change for new hardware.
- Software written for specific internal hardware.
- Third party software tied to specific hardware or operating system.
- Hardware drivers tied to specific operating system
- Relying on timing or characteristics of hardware interface for timing issues.

Solution:

- Use standard drivers that will be maintained by the operating system.
- Only use standard calls or better create a library of standard functions that uses standard calls.
- Consolidate all GUI functions into a Host piece of software so that changes to the GUI affected by a new operating system can be fixed in one place.
- Use only interfaces that are likely to be supported in future versions of the operating system.
- The only timing issues that should be addressed in the host computer software is that the data stream is fast enough to keep up with the device being loaded. The external hardware should take care of the timing and provide data flow control.

Selection of Standard PC Interface

RS-232

- Too slow
- May not be available on new computers
- Standard data protocol not defined

Parallel Port

- May not be available on new computers
- Standard data protocol not defined
- Significant differences between computers

SCSI

- Fast
- Data protocol well defined
- Not enough ruggedized computers have this interface built in.

Firewire

- Fast
- Data protocol well defined
- Good power source 12 Volts at
- Some ruggedized computers have this interface

Universal Serial Bus (USB)

- Fast
- Data protocol well defined
- 2.5W power available
- Standard on most ruggedized computers.